

Model Confidence Bounds for Variable Selection

Yang Li, Zhibing He, Yuetian Luo, Davide Ferrari and Yichen Qin*

November 30, 2016

Abstract

We introduce the model confidence bounds (MCBs) for variable selection in the context of nested parametric models. Similarly to the endpoints in the familiar confidence interval for parameter estimation, the MCBs identify two nested models (upper and lower confidence bound models) containing the true model at a given level of confidence. Instead of trusting a single selected model by a given selection method, the MCBs width and composition enable the practitioner to assess the overall model uncertainty. The MCBs methodology is implemented by a fast bootstrap algorithm which is shown to yield the correct asymptotic coverage under rather general conditions. A new graphical tool – the model uncertainty curve (MUC) – is introduced to visualize the variability of model selection and compare different model selection procedures. Our

*Yang Li is Associate Professor, Center for Applied Statistics and School of Statistics, Renmin University of China, Beijing, China. Zhibing He and Yuetian Luo are Research Assistants, Statistical Consulting Center, Renmin University of China. Davide Ferrari is Associate Professor, School of Mathematics and Statistics, University of Melbourne, Australia. Yichen Qin is Assistant Professor, Department of Operations, Business Analytics and Information Systems, University of Cincinnati, Cincinnati, OH, USA. Yichen Qin is the corresponding author: qinyin@ucmail.uc.edu. This work is supported by the Research Funds of Renmin University of China under Grant 15XNI011.

Monte Carlo simulations and real data examples confirm the validity and illustrate the advantages of the proposed method.

Keywords: confidence set, model selection, nonparametric bootstrap.

1 Introduction

Variable selection is an important and well-studied topic. Many modern analyses aim at selecting a subset of variables from a very large number of predictors, while attempting to attenuate possible modeling bias. In the context of linear (and generalized linear) models a wealth of methods have been introduced to enhance predictability and to select significant predictors. These include popular sparsity-inducing penalization methods such as the Lasso Tibshirani (1996), SCAD Fan and Li (2001); Yang et al. (2011), elastic net Zou and Hastie (2005); De Mol et al. (2009), adaptive Lasso Zou (2006); Zhang and Lu (2007); Huang et al. (2008), group Lasso Yuan and Lin (2006); Meier et al. (2008); Simon et al. (2013) and mini-max convex penalization (MCP) Jiang et al. (2013).

Regardless of the selection procedure used, variable selection uncertainty is an important and ubiquitous aspect of the variable selection activity. Often using the same variable selection method on different samples from a common population results different models. Even for a single sample, different variable selection methods tend to select different sets of variables in the presence of pronounced noise. Motivated by the need to address this model ambiguity, there has been a growing interest in developing model confidence set (MCS) methods, which may be broadly regarded as frequentist approaches to obtain a set models statistical equivalent to an optimal model at a certain level of confidence $1 - \alpha$. A MCS extends the familiar notion of confidence intervals to the model-selection framework and enables one to assess the uncertainty associated with a given selection procedure. If the data are informative, the MCS contains only a few models (exactly one model in the case of

overwhelming information), while uninformative data correspond to a large MCS.

Hansen et al. (2011) propose to construct a MSCs from a given set of candidate model by a sequence of equivalence test on the currently remaining models followed by an elimination rule to remove the worst model. Their method obtains a subset of the original models that is meant to contain (or equal) the set of models with the best performance under some given loss function. Ferrari and Yang (2015) introduce the notion of variable selection confidence set (VSCS) for linear regression. While sharing the same motivation with Hansen et al. (2011), their method constructs the MCS by a sequence of F-tests and achieves exact coverage probability for the globally optimal model without necessarily relying on a user-defined initial list of models. They show that without restrictions on the model structure (e.g. sparsity), the size of the VSCS is potentially large, thus reflecting the possible model selection uncertainty. To address this issue, they introduce the notion of lower bound models (LBMs) – i.e. the most parsimonious models that are not statistically significantly inferior to the full model at a given confidence level – and study their properties. Previously, Shimodaira (1998) advocate the use of a set of models that have AIC values close to the smallest among the candidates based on hypothesis testing. Hansen et al. (2003, 2005) apply a MCS procedure in the context of volatility and forecasting models. Samuels and Sekkel (2013) use the MCS to a subset of models prior to averaging the resulting forecasts.

The methods above yield confidence sets satisfying a nominal coverage probability for an optimal model. However, the models contained are not constrained in terms of their structure, meaning that models in a confidence sets may be drastically different in their composition with no common variables. This poses some challenges in interpreting the models in a confidences set. In this paper, we introduce a new procedure which computes the so-called model confidence bounds (MCBs). The MCBs are constructed by finding a larger and a smaller model – called lower bound model (LBM) and upper bound model (UBM), respectively – which are nested and such that true model is included between those two

at user-specified confidence level $1 - \alpha$. The upper and lower bounds of our MCBs have a rather natural interpretation: The LBM is regarded as the most parsimonious model containing indispensable predictors, while models containing variables beyond the UBM include superfluous predictors.

Our methodology provides a platform to assess the uncertainty associated with different selection methods. Through comparing the width and composition MCBs of different methods, the practitioner can decide which method yield more stable results. The MCBs can also be used as a model selection diagnostic tool. If a model selected is not within the MCBs at a certain confidence level, there is a strong reason to doubt the soundness of its predictors. The proposed method is based on nonparametric bootstrap, so it does not rely on the parametric distribution assumptions and may be applied to a wide range of model families. Finally, to calculate the MCBs, we first propose an exact but computationally intensive, algorithm; we further propose a much more efficient approximated algorithm which is found to have a performance comparable to that of the exact algorithm.

The rest of the paper is organized as follows. In Section 2, we describe our methodology: we introduce the optimal MCBs and describe bootstrap method that computes the MCBs from the data. In the same section, we develop numerical and graphical summaries to assess model uncertainty. In Section 3, we present our algorithms to find approximate MCBs and introduce a criterion. In Section 4, we apply the proposed method to a real data set. In Section 5, we carry out Monte Carlo experiments to study the coverage and the numerical performance of our algorithm. Finally, in Section 6 we conclude and give final remarks.

2 Methodology

2.1 Preliminaries

For concreteness, we focus on generalized linear regression models, but our methodology can be applied to any model with a similar nested structure. Let Y be a $n \times 1$ response vector with mean $\boldsymbol{\mu} = E(Y)$ and such that

$$g^{-1}(\boldsymbol{\mu}) = X\boldsymbol{\theta}, \quad (1)$$

where $g(\cdot)$ is a continuous invertible link function, X is a $n \times p$ matrix of predictors with i th row vector \mathbf{x}_i , $\boldsymbol{\theta} \in \mathbb{R}^p$ is the parameter vector. We assume Y in a given set of distributions \mathcal{F} . For example, \mathcal{F} might be the set of all n -variate normal distributions with independent components with equal variances, then Y follows $N_n(X\boldsymbol{\theta}, \sigma^2 I_n)$.

Some of the predictors in X are unimportant in terms of explaining Y , in the sense that some of the elements in $\boldsymbol{\theta} = (\theta_1, \dots, \theta_p)^T$ are zeros, but we do not know which ones are unimportant. Let \hat{m} be the index set of some predictors in the sense that it defines a possible model and \mathcal{M}_{all} denotes a set of all the possible models \hat{m} . Then we have the following definition:

Definition 1. *Let m^* be the index set of predictors with non-zero true coefficients,*

$$m^* = \{j : \theta_j \neq 0, j = 1, \dots, p\}.$$

Let m_{full} be the index set of all predictors,

$$m_{full} = \{j : j = 1, \dots, p\}.$$

Therefore, m^* represents the true model, and m_{full} represents the full model with all

predictors.

Without loss of generality, we assume that the first p^* coefficients in $\boldsymbol{\theta}$ are different from zero, so that the true model $m^* = \{1, \dots, p^*\}$. In certain situations, prior information on the model structure enables us to restrict further the set of all possible models \mathcal{M}_{all} . For example, in polynomial regression one would include a certain power of the predictor only if all lower order terms appear as well. Another case is when certain predictors are always protected in the sense that they appear in all candidate models.

Since \hat{m} represents a subset of all predictors, here we focus on penalized likelihood selection methods; specifically, $\hat{m} = \{j : \hat{\theta}_j \neq 0\}$, $1 \leq j \leq p$ where the estimator $\hat{\boldsymbol{\theta}}$ minimizes a penalized likelihood criterion with the form

$$\ell_n(\boldsymbol{\theta}) = -2 \log L_n(\boldsymbol{\theta}; Y, X) + \lambda_n \rho(\boldsymbol{\theta}), \quad (2)$$

where $L_n(\boldsymbol{\theta}; Y, X)$ is the likelihood function, $\lambda_n \geq 0$ is a user-defined regularization parameter and $\rho(\boldsymbol{\theta})$ is some penalty function $\rho : \mathbb{R}^p \mapsto \mathbb{R}^+$. Throughout the paper $\rho(\boldsymbol{\theta})$ will be a type of norm. For example, $\rho(\boldsymbol{\theta}) = \sum_{j=1}^p I(\theta_j \neq 0)$ corresponds to the L_0 -norm and yields a number of information theoretical selection criteria, including the Akaike information criterion (AIC), for $\lambda = 2$, and the Bayesian information criterion (BIC), for $\lambda = \log(n)$. Setting $\rho(\boldsymbol{\theta}) = \sum_{j=1}^p |\theta_j|$ gives the L_1 -norm which corresponds to Lasso estimation.

2.2 Model Confidence Bounds

For a given sample $D = \{(y_i, \mathbf{x}_i), i = 1, \dots, n\}$, where \mathbf{x}_i is a p -dimensional vector, from the regression model (1), we want to find a small model $\hat{m}_L = \hat{m}_L(D)$ and a large model $\hat{m}_U = \hat{m}_U(D)$ such that the true model m^* is nested between \hat{m}_L and \hat{m}_U with probability at least $1 - \alpha$. Note that $\hat{m}_L \subseteq \hat{m}_U$.

Definition 2 (Model confidence set and bounds). *The $100(1 - \alpha)\%$ -model confidence bounds*

(MCBs) are defined by the pair of models $\{\hat{m}_L, \hat{m}_U\}$ such that

$$P(\hat{m}_L \subseteq m^* \subseteq \hat{m}_U) \geq 1 - \alpha. \quad (3)$$

The models \hat{m}_L and \hat{m}_U are called the lower bound model (LBM) and the upper bound model (UBM), respectively. The $100(1 - \alpha)\%$ -model confidence set (MCS) is defined

$$\widehat{\mathcal{M}}_\alpha = \{\hat{m} : \hat{m}_L \subseteq \hat{m} \subseteq \hat{m}_U\} \quad (4)$$

If (3) is valid as $n \rightarrow \infty$, $\{\hat{m}_L, \hat{m}_U\}$ define asymptotic model confidence bounds (AMCBs) and asymptotic model confidence set (AMCS).

The above definition extends the usual notion of confidence interval to the variable selection setting. Similarly to the familiar confidence interval for a population parameter, MCBs cover the true model m^* with a certain probability. A model smaller than \hat{m}_L , is regarded as too parsimonious in the sense that it is likely to miss at least one important variable, whilst models with the variables in \hat{m}_U plus other predictors are considered to be over-fitting. Similarly to the familiar confidence interval for parameter estimation, one can obtain a one-sided $100(1 - \alpha)\%$ -MCBs by setting $\hat{m}_L = \emptyset$ or $\hat{m}_U = m_{\text{full}}$.

The pair of models $\{\hat{m}_L, \hat{m}_U\}$ which represent two extreme models, i.e. the most parsimonious and complex models yet not significantly different from the true model m^* . Using these two models, we can list all possible models nested between those two extremes, resulting in an easy-to-interpret hierarchical structure. Moreover, the size of the MCBs reflects the model selection uncertainty in a given sample. When the amount of information in the data is very large, the MCBs will contain only a few models. In the extreme case of overwhelming information (e.g., fixed p , $n \rightarrow \infty$) the MCBs contain only the true model so that $\hat{m}_L = \hat{m}_U = m^*$. In most practical situations $\hat{m}_L \subset \hat{m}_U$ with the discrepancy between the

size of the lower and upper bound models becoming large when the data are uninformative.

Since there are usually multiple MCBs satisfying Equation (3), in practice we will only consider the MCBs with smallest width.

Definition 3 (Optimal MCBs). *Let $w(\hat{m}_L, \hat{m}_U) = |\hat{m}_U| - |\hat{m}_L|$ be the width associated with the bounds $\{\hat{m}_L, \hat{m}_U\}$, where $|A|$ represents the cardinality of set A . The $100(1 - \alpha)\%$ -MCBs $\{\hat{m}_L, \hat{m}_U\}$ are said to be optimal if $w(\hat{m}_L, \hat{m}_U) \leq w(\hat{m}'_L, \hat{m}'_U)$ for all MCBs $\{\hat{m}'_L, \hat{m}'_U\}$.*

Therefore, for any given confidence level, optimal MCBs have the shortest width among all possible confidence bounds.

2.3 Bootstrap Construction

Given the data set $D = \{(y_i, \mathbf{x}_i), i = 1, \dots, n\}$, we generate B bootstrap samples $D^{(b)} = \{(y_i^{(b)}, \mathbf{x}_i^{(b)}), i = 1, \dots, n\}$, $b = 1, \dots, B$. Then, we obtain the set of bootstrap models $\mathcal{M}_{boot, B} = \{\hat{m}^{(b)}, b = 1, \dots, B\}$ by applying a model selection method to each bootstrap sample $D^{(b)}$. For any two nested models, $m_1 \subseteq m_2$ (m_1 and m_2 denote the index set of some predictors), it seems quite natural to estimate the probability of the event $\{m_1 \subseteq m^* \subseteq m_2\}$ using the following statistic.

Definition 4. *The bootstrap coverage rate (BCR) of models $m_1 \subseteq m_2$ is*

$$\hat{r}(m_1, m_2) = \frac{1}{B} \sum_{b=1}^B I(m_1 \subseteq \hat{m}^{(b)} \subseteq m_2), \quad (5)$$

where $\hat{m}^{(1)}, \dots, \hat{m}^{(B)}$ are bootstrap models and $I(\cdot)$ is the indicator function.

To obtain the upper and lower bound models $\hat{m}_L \subseteq \hat{m}_U$, we solve the empirical objective

$$(\hat{m}_L, \hat{m}_U) = \underset{m_1, m_2}{\operatorname{argmin}} \{w(m_1, m_2) \text{ s.t. } \hat{r}(m_1, m_2) \geq 1 - \alpha\}. \quad (6)$$

For a given α , we find a pair of nested models as a solution to the approximate equation $\hat{r}(m_1, m_2) = 1 - \alpha$. It is clear that this approach works as long as the bootstrap coverage rate estimates consistently the nominal coverage $1 - \alpha$. A more detailed discussion on this issue is deferred to Section 2.5.

As an illustration, consider the 90%-MCBs obtained for the linear model $Y = X\theta + \epsilon$, with $p = 6$ and $\epsilon \sim N_n(0, 1)$ based on $n = 30, 150, 3000$ samples. The model space \mathcal{M}_{all} contains $64 = 2^6$ unique models and the true model is m^* corresponds to coefficients $\theta^* = (1, 1, 1, 0, 0, 0)$. We generate $B = 1000$ bootstrap models selected by the adaptive lasso selection method with tuning constant selected by 10-fold cross-validation Huang et al. (2008). Figure 1 depicts all the possible models along with 90%-MCBs with the horizontal axis denoting the number of predictors. Sampled and not sampled models are denoted respectively by solid and empty circles. Crossed models lay between the 90%-MCBs, and thus denote the models in the 90%-MCS. Models are plotted from bottom to top to in descending order according to their frequency in the bootstrap set of models $\mathcal{M}_{boot, B}$.

Note that the true model is always contained between the MCBs, and the models \hat{m}_L , \hat{m}_U and m^* have the largest frequency since they appear at the bottom of the graph. Moreover, similarly to the familiar confidence intervals in parameter estimation, as n increases, there are fewer models in the MCBs, which become increasingly similar to true model. For very large n , we have $\hat{m}_L = \hat{m}_U = m^*$, which confirms the expected theoretical behaviour described in the next section (see Figure 1 (c)).

2.4 Graphical Assessment of Selection Uncertainty

The MCBs methodology can be used to assess the amount of model selection uncertainty associated with a given model selection method. Let $\hat{m}^{(1)}, \dots, \hat{m}^{(B)}$ be bootstrapped models

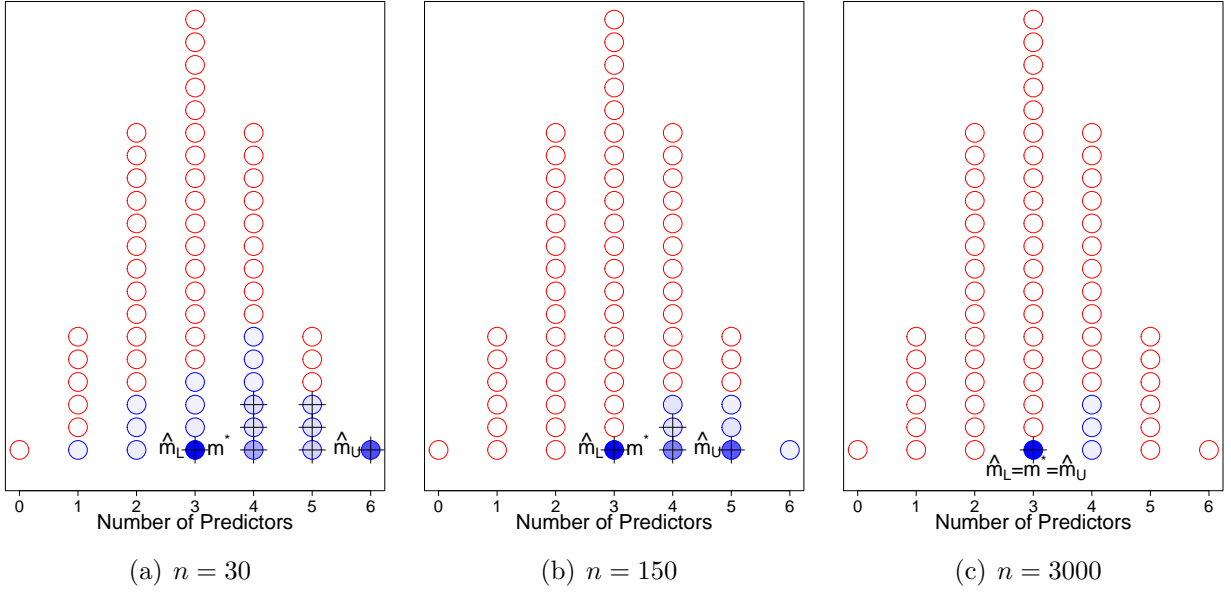


Figure 1: 90% MCBs and MCS for different sample sizes. Sampled and not sampled models are denoted by the solid and empty circles, respectively; the crossed circles denote models nested between the MCBs (i.e. in the MCS). The horizontal axis denotes the number of predictors and models are plotted from bottom to top in descending order according to their bootstrap frequency. The lower and upper bound models (\hat{m}_L and \hat{m}_U) and true model (m^*) are also marked. Data are generated from the normal regression model $Y \sim N_n(X\theta, I_n)$ with $\theta = (1, 1, 1, 0, 0, 0)^T$; the plot is based on 1000 bootstrap models obtained by 10-fold cross-validated Adaptive Lasso.

under some model-selection method. The profiled bootstrap coverage rate is

$$\text{CR}(w) = \hat{r}(\hat{m}_L, \hat{m}_U) = \frac{1}{B} \sum_{b=1}^B I(\hat{m}_L \subseteq \hat{m}^{(b)} \subseteq \hat{m}_U). \quad (7)$$

The CR statistic is essentially a function of the MCBs width $w = |\hat{m}_U| - |\hat{m}_L|$ since \hat{m}_L and \hat{m}_U implicitly depend on w . Ideally, one would like to use the nominal coverage probability $1 - \alpha \simeq P(\hat{m}_L \subseteq m^* \subseteq \hat{m}_U)$ as a measure of uncertainty for a variable selection method. However, in practice the exact rate is unknown and instead the CR statistic is used to approximate $1 - \alpha$. When a consistent model selection method is used (BIC, Adaptive Lasso, MCP, SCAD) the nominal coverage $1 - \alpha$ and the estimated rate CR are typically very close (see Section 5.4).

Clearly, a good model selection method would tend to return MCBs with a larger coverage value at a given width, or a lower width at a given coverage. Thus we propose to assess the uncertainty of a given model-selection mechanism by plotting the pairs

$$\mathcal{P}_{MUC} = \{(w/p, \text{CR}(w)), 0 \leq w \leq p\}.$$

The resulting plot is called a model uncertainty curve (MUC). The MUC of a given variable selection method with good performance will tend to arch towards the upper left corner. The MUC curve is in some sense analogous to that of a receiver operating characteristic (ROC) curve used to assess binary classifiers. The ideal model selection method has $w/p = 0$ and $\text{CR}(0) = 1$, i.e. no model selection uncertainty at all and perfect coverage (top left corner of the plot). Moreover, the area under the MUC (AUC) can be used as a raw measure of uncertainty for the variable selection method under exam. A larger value of AUC implies less uncertainty and more stability of the corresponding variable selection method. Overall, we can decide which method has the best performance according to the shape of the MUC

and the corresponding AUC.

2.5 Asymptotic Coverage

The quality of the selected model \hat{m} in terms of underfitting and overfitting probabilities, i.e. the probability of the events

$$\mathcal{M}_n^- = \{\hat{m} \not\supseteq m^*\}, \quad \mathcal{M}_n^+ = \{\hat{m} \supsetneq m^*\}, \quad (8)$$

Note that \mathcal{M}_n^+ represents the case of overfitting, i.e. the terms in the true model are selected plus some additional superfluous terms. The set \mathcal{M}_n^- represents underfitting, i.e. some terms in the true model are missed. A model selection procedure is consistent if $P(\hat{m} = m^*) \rightarrow 1$ as $n \rightarrow \infty$ for every $\theta \in \mathbb{R}^p$, which occurs if $P(\mathcal{M}_n^-) \rightarrow 0$ and $P(\mathcal{M}_n^+) \rightarrow 0$ as $n \rightarrow \infty$. If $P(\mathcal{M}_n^-) \rightarrow 0$, but \hat{m} is not consistent then we say that the procedure is conservative.

Proposition 1. *Assume: (A.1) (Model selection consistency) $P(\mathcal{M}_n^\pm) = o(1)$; (A.2) (Bootstrap validity) For the re-sampled model $\hat{m}^{(b)}$, assume $P(\hat{m}^{(b)} \neq \hat{m}) = o(1)$. Then, for \hat{m}_L and \hat{m}_U solving program (6), we have*

$$P(\hat{m}_L \subseteq m^* \subseteq \hat{m}_U) \geq 1 - \alpha + o(1). \quad (9)$$

Proof. The result follows by a straightforward application of the law of total probability. Let $\mathcal{A} = \{\hat{m}_L \subseteq m^* \subseteq \hat{m}_U\}$ be the event that the true model m^* is nested between the lower and upper bound models. Then

$$\begin{aligned} P(\mathcal{A}^c) &= P(\mathcal{A}^c | \hat{m} = m^*)P(\hat{m} = m^*) + P(\mathcal{A}^c | \hat{m} \neq m^*)P(\hat{m} \neq m^*) \\ &\leq P(\{\hat{m}_L \subseteq \hat{m}\}^c \cup \{\hat{m} \subseteq \hat{m}_U\}^c) + o(1) \\ &\leq P(\hat{m}_L \not\supseteq \hat{m}) + P(\hat{m}_U \not\supseteq \hat{m}) + o(1), \end{aligned}$$

where the probabilities in the last expression concern the event that \hat{m}_L overestimates m^* and \hat{m}_U underestimates \hat{m}_L . Let $\hat{m}^{(b)}$ denote a bootstrapped model and note that

$$\begin{aligned} P(\hat{m}_L \supsetneq \hat{m}) &= P(\hat{m}_L \supsetneq \hat{m} | \hat{m}^{(b)} \not\subseteq \hat{m}_L) P(\hat{m}^{(b)} \not\subseteq \hat{m}_L) \\ &\quad + P(\hat{m}_L \supsetneq \hat{m} | \hat{m}^{(b)} \subseteq \hat{m}_L) P(\hat{m}^{(b)} \subseteq \hat{m}_L) \\ &\leq P(\hat{m}^{(b)} \not\subseteq \hat{m}_L) + P(\hat{m}^{(b)} \supsetneq \hat{m}) \end{aligned} \quad (10)$$

Similarly, $P(\hat{m}_U \not\supseteq \hat{m}) \leq P(\hat{m}^{(b)} \supsetneq \hat{m}_U) + P(\hat{m}^{(b)} \not\supseteq m^*)$. Combining this with (10) implies

$$P(\mathcal{A}^c) \leq P(\hat{m}^{(b)} \not\subseteq \hat{m}_L) + P(\hat{m}^{(b)} \supsetneq \hat{m}_U) + P(\hat{m}^{(b)} \supsetneq m^*) + P(\hat{m}^{(b)} \not\supseteq \hat{m}) + o(1) \quad (11)$$

$$\leq \alpha + 2P(\hat{m}^{(b)} \neq \hat{m}) + o(1). \quad (12)$$

The desired result follows from Assumption A.2 □

The above proposition shows that if the model-selection procedure is consistent (i.e. the probability of overfitting or underfitting the underlying) model becomes small as the sample size increases), then the BCR statistic estimates consistently the true coverage probability associated with models $m_1 \subseteq m_2$.

We conclude this section by discussing the important case of linear models with L_1 -penalty $\lambda_n \rho(\theta) = \lambda_n \sum_{j=1}^p |\theta_j|$. Yang and Yang (2015) show that Condition A.1 (model consistency) is satisfied under the so-called restricted eigenvalue condition.

3 Algorithms

3.1 Naive Implementation by Exhaustive Search

While traditional confidence intervals for parameter estimation are typically computed by finding lower and upper bounds based on a given confidence level, our MCBs are determined

in a reverse way. Specifically, for given widths $w = 0, 1, \dots, p$ we obtain a sequence of MCBs and then choose the MCBs with the smallest bootstrap coverage rate (5) that is no smaller than the nominal confidence level $1 - \alpha$. The MCBs obtained in this way is interpreted as the one with the least width for a given confidence level. This straightforward procedure is detailed in our first algorithm.

Algorithm 1: Naive $100(1 - \alpha)\%$ -model confidence bound

Inputs: Data $D_n = \{(\mathbf{x}_i, y_i), i = 1, \dots, n\}$; Confidence level $1 - \alpha$.

Output: MCBs for given confidence level $1 - \alpha$.

1. Generate B bootstrap samples $D_n^{(b)} = \{(x_i^{(b)}, y_i^{(b)}), i = 1, \dots, n\}$, $b = 1, \dots, B$ and obtain corresponding models $\hat{m}^{(1)}, \dots, \hat{m}^{(B)}$ using a consistent model selection method.
2. **For** widths $w = 0, \dots, p$ obtain confidence bounds by solving

$$\{\hat{m}_L(w), \hat{m}_U(w)\} = \max_{m_1 \subseteq m_2} \{\hat{r}(m_1, m_2) \text{ s.t. } |m_1| - |m_2| = w\},$$

where \hat{r} is the bootstrap empirical coverage rate defined in (5).

3. From the MCBs in Step 2 obtain the optimal MCBs $\{\hat{m}_L^*, \hat{m}_U^*\} = \{\hat{m}_L(w^*), \hat{m}_U(w^*)\}$, where

$$w^* = \underset{0 \leq w \leq p}{\operatorname{argmin}} \{|\hat{m}_U(w)| - |\hat{m}_L(w)| \text{ s.t. } \hat{r}(\hat{m}_L(w), \hat{m}_U(w)) \geq 1 - \alpha\}.$$

Next we explore the computational cost associated with Algorithm 1. The number of operations involved by Step 2 of Algorithm 1 for the fixed width w is

$$\Omega_1(w) = \binom{p}{0} \binom{p}{w} + C_p^1 C_{p-1}^w + \dots + C_p^{p-w} C_w^w = \frac{p!}{w!} \sum_{k=0}^{p-w} \frac{1}{k!(p-w-k)!}.$$

Then the total number of operations is obtained as

$$\Omega_1 = \sum_{w=0}^p \Omega_1(w) = p! \sum_{w=0}^p \sum_{k=0}^{p-w} \frac{1}{k!w!(p-k-w)!}.$$

Although Algorithm 1 returns an exact solution for the optimal MCBs described in (6), this naive strategy essentially requires exhaustive enumeration and it is therefore applicable only in cases where p is very small. Thus, next we turn our interest to another strategy which achieves similar accuracy in terms of coverage, while involving a much reduced computational burden.

3.2 Implementation by Predictor Importance Ranking

The theoretical findings in Section 2.5 show that the predictors in the true model are selected with large frequency compared to that of irrelevant predictors. This is illustrated by the numerical example in Figure 1. Thus an appropriate MCBs is highly correlated with the order of predictors based on their selected times. Thus we propose Algorithm 2. The more selected times the predictor, the more likely it is to be included in \hat{m}_L or \hat{m}_U .

Consider the relative frequency of the j th predictor among B bootstrapped models. Specifically, the importance of predictor j is measured by the frequency $\bar{\pi}_j = B^{-1} \sum_{b=1}^B I(j \in \hat{m}^{(b)})$, $j = 1, \dots, p$, with $\hat{m}^{(b)}$ is a bootstrapped model in $\mathcal{M}_{boot, B}$. Let $\Pi = (u_1, \dots, u_p)$ be the arrangement of indexes in $\{1, \dots, p\}$ induced by the ordered frequencies $\bar{\pi}_{u_1} > \dots > \bar{\pi}_{u_p}$ (assuming no ties). The ordering Π induces a natural ranking of predictors in terms of their selection frequency. For example, predictor x_{u_p} is selected the least times in the B bootstrapped models, whilst x_{u_1} is the one selected most frequently.

For a fixed MCBs width w , we consider constructing the lower confidence bound model \hat{m}_L by taking the k most important predictors according to the ordering Π , where $k \leq p - w$ is to be specified. The upper confidence bound model \hat{m}_U is constructed by taking a few others until reaching the desired width w , i.e

$$\hat{m}_L(k) = \{u_1, \dots, u_k\}, \quad \hat{m}_U(k, w) = \{u_1, \dots, u_k, u_{k+1}, \dots, u_{k+w}\} \quad (13)$$

Thus, given the width $0 \leq w \leq p$, lower and upper confidence bound models are obtained as

$$\{\hat{m}_L, \hat{m}_U\} = \operatorname{argmax}_{k \leq w} \hat{r}(\hat{m}_L(k), \hat{m}_U(k, w)), \quad (14)$$

where \hat{r} is the bootstrap coverage rate defined in (5).

According to the previous theoretical and empirical results, MCBs are highly related to the order of predictors based on their selected times. It makes sense that for a given width w the \hat{m}_L and \hat{m}_U include the first k and $k + w$ predictors based on the order of predictors, respectively. As for how to decide the value of k , $k \leq p - w$, it needs to compare all the possible MCBs $\{\hat{m}_L, \hat{m}_U\}$ and choose the MCBs which have the maximum \hat{r} .

The above discussion leads to our second algorithm obtained by modifying Steps 3 and 4 of Algorithm 1. Since the construction of the lower and upper bound models are driven by the ordering of predictors in terms of their estimated importance, we call this new algorithm $1 - \alpha$ -MCBs construction by predictor importance (PI) ranking or simply PI-MCBs.

Algorithm 2: $1 - \alpha$ -model confidence bound by predictor importance ranking

Inputs: Data $D_n = \{(\mathbf{x}_i, y_i), i = 1, \dots, n\}$; Confidence level $1 - \alpha$.

Output: MCBs for given confidence level $1 - \alpha$.

1. Same as Step 1 of Algorithm 1.
2. **For** each $j = 1, \dots, p$, predictor importance $\bar{\pi}_j = B^{-1} \sum_{b=1}^B I(j \in \hat{m}^{(b)})$, where $\hat{m}^{(b)}$, $b = 1, \dots, B$ are bootstrapped models obtained in Step 2. Then generate the ordering $\Pi = \{u_1, \dots, u_p\}$ induced by $\bar{\pi}_{u_1} > \dots > \bar{\pi}_{u_p}$.
3. **For** width $w = 0, 1, \dots, p$ and $k \leq p - w$, compute lower and upper bound models $\hat{m}_L(k) = \{u_1, u_2, \dots, u_k\}$ and $\hat{m}_U(k, w) = \{u_1, u_2, \dots, u_{k+w}\}$. The MCBs have bounds $\{\hat{m}_L(k^*), \hat{m}_U(k^*, w)\}$ where

$$k^* = \operatorname{argmax}_{k \leq p-w} \{\hat{r}(\hat{m}_L(k), \hat{m}_U(k, w))\}.$$

4. From the MCBs in Step 3 obtain the optimal $1 - \alpha$ MCBs $\{\hat{m}_L^*, \hat{m}_U^*\} = \{\hat{m}_L(w^*), \hat{m}_U(w^*)\}$, where

$$w^* = \operatorname{argmin}_{0 \leq w \leq p} \{|\hat{m}_U(w)| - |\hat{m}_L(w)| \text{ s.t. } \hat{r}(\hat{m}_L(w), \hat{m}_U(w)) \geq 1 - \alpha\}.$$

The PI-MCBs Algorithm is extremely fast compared to Algorithm 1. In particular, the number of operations required by Algorithm 2 can be expressed as

$$\Omega_2 = 1 + \sum_{w=0}^{p-1} (p - w) = \frac{p^2 + p + 2}{2}.$$

Figure 2 compares the the computational cost of Algorithms 1 and 2 in terms of their number of operations required (on the log scale for comparison purpose). The number of operations required by Algorithm 2 grow slowly compared to Algorithm 1. For example when $p = 10$ Algorithms 1 and 2 require about 60,000 and 60 operations, respectively. On the other hand the number of operations of Algorithm 1 increases very rapidly in the number of predictors p , suggesting that this is not a viable algorithm for high-dimensional regression problems where p is large.

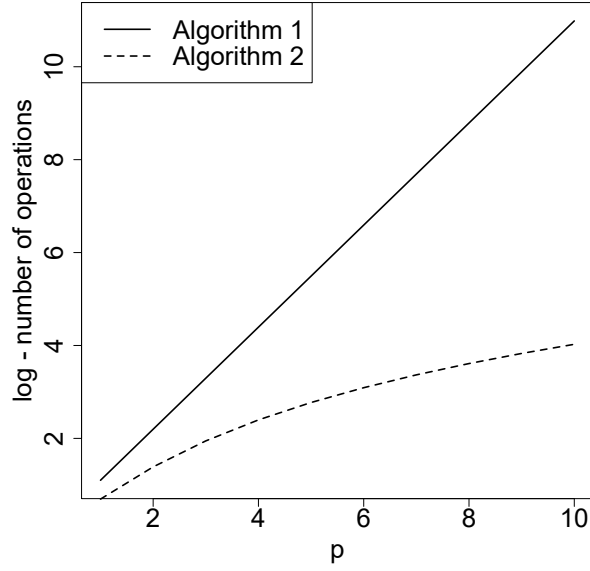


Figure 2: Logarithmic numbers of operations of Algorithm 1 and Algorithm 2 for different p . The total operations of both Algorithm 1 and Algorithm 2 greatly depend on the Step 2 which needs some loop iterations. Specifically, in Algorithm 1 even for a given width w ($0 \leq w \leq p$), to find the exhaustive candidate models in \hat{m}_L and \hat{m}_U it often needs substantial numbers of operations. Whilst in Algorithm 2 it needs at most $p - w$ iterations for a given w .

4 Real Data Analysis

The Diabetes data Efron et al. (2004) consists of measurements on $n = 442$ diabetic patients. The response is a measurement of disease progression one year after baseline, and ten predictors: body mass index (**bmi**), lamotrigine (**ltg**), mean arterial blood pressure (**map**), total serum cholesterol (**tc**), sex (**sex**), total cholesterol (**tch**), low- and high-density lipoprotein (**ldl** and **hdl**), glucose (**glu**) and age (**age**). To construct the MCBs with Algorithm 2 at different confidence levels, we generate 1000 bootstrap models using the following methods: Adaptive Lasso, Lasso and Stepwise (BIC).

In Figure 3 we compare the methods using the model uncertainty curve (MUC) introduced in Section 2.4. 1000 samples are bootstrapped from the original dataset. Note that the coverage rate (CR) increases with the confidence set width w ; when $w < 2$ the coverage rate is 0.1, which means that the corresponding MCBs capture only 100 of bootstrap models. When $w > 6$ the bootstrap coverage is already larger than 0.9 meaning that the MCBs contain more than 900 bootstrap models, which makes this curve arch towards the upper left of the figure. Recall that the interpretation of the MUC curve is similar to that of the more familiar ROC curve.

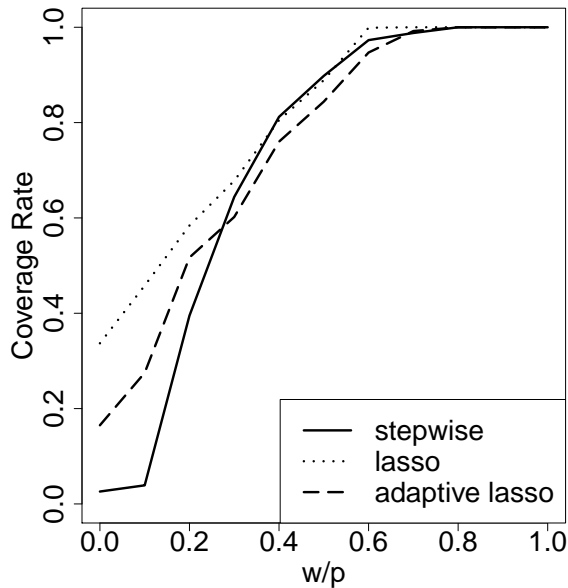


Figure 3: Performance of MUC in Diabetes. 1000 samples are bootstrapped from the original dataset. Stepwise selects variables based on BIC and tuning parameters of other two methods are chosen by 10-fold cross validation.

Table 1 shows upper and lower bounds for 95%- and 75%-MCBs. Note that when the confidence level increases, the LBM becomes smaller while the UBM is gets larger. Since for a given confidence level $1 - \alpha$ we choose the MCBs which have the smallest CR that no less than $1 - \alpha$, the CR is usually larger than the confidence level for a given MCBs. Note that for 95% MCBs, comparing with other two methods Lasso tends to select more predictors and therefore it has larger lower bound. In addition, the 95% MCBs contain all the predictors which are selected by the corresponding method for the full dataset. According to our MCBs procedure, the predictors (**bmi**, **ltg**, **map** are considered most indispensable which is consistent with other existing studies Lindsey et al. (2010); Efron et al. (2004).

Method	$1 - \alpha$	MCBs	bmi	ltg	map	tc	sex	tch	ldl	glu	hdl	age	width	CR
ALasso	95%	UBM	X	X	X	X	X	X	X	X	X		6	97.5%
		LBM	X	X	X									
	75%	UBM	X	X	X	X	X	X	X	X			4	81.1%
		LBM	X	X	X	X								
Lasso	95%	UBM	X	X	X	X	X	X	X	X	X		5	99.9%
		LBM	X	X	X	X								
	75%	UBM	X	X	X	X	X	X	X	X	X		4	80.5%
		LBM	X	X	X	X	X							
Stepwise	95%	UBM	X	X	X	X	X	X	X	X	X		6	97.3%
		LBM	X	X	X									
	75%	UBM	X	X	X	X	X	X	X	X			5	81.2%
		LBM	X	X	X									
ALasso			X	X	X	X	X	X	X					
Lasso			X	X	X	X	X	X		X	X			
Stepwise			X	X	X		X				X			

Table 1: MCBs of different variable selection methods at 75% and 95% confidence levels. “X” denotes the predictor is in a LBM or a UBM, and ALasso represents Adaptive Lasso. For the original data set, 1000 bootstrap samples are generated. Adaptive Lasso, Stepwise and Lasso are used to select important predictors for each sample respectively. The last row shows the results of applying these three methods to the full dataset.

5 Numerical Studies

In this section, we investigate the performance of the proposed method by Monte Carlo (MC) experiments. Each MC sample is generated from the model

$$y_i = \sum_{j=1}^{p^*} x_{j,i} + \sum_{j=p^*+1}^p x_{j,i} \times 0 + \epsilon_i, \quad \epsilon_i \sim N_1(0, \sigma^2). \quad (15)$$

where p^* is the size of the true model, and each $x_i = (x_{1,i}, \dots, x_{p,i})$ is sampled from a n -variate normal distribution $N_n(0, \Sigma)$ at each MC run. The ij th elements of the covariance matrix Σ are $\Sigma_{ij} = \rho^{|i-j|}$, $0 \leq \rho < 1$, with $\rho = 0$ corresponding to the case of orthogonal predictors. The main goals of the following experiment is to evaluate the performance of

the three algorithms detailed in Section 3 and assess the statistical accuracy in terms of coverage probability. Additional numerical illustrations will focus on the comparison of different variable selection methods by our MCBs approach.

5.1 Comparison of Algorithms for MCBs Construction

We consider the same model as Equation (15) and six scenarios: $p = 8, 10, 15, 50, 100, 200$ respectively. In each scenario $\rho = 0$, $n = 300$, and $B = 1000$. While in the first three scenarios $\sigma = 1$ and $\sigma = 0.3$ for the last three scenarios. Adaptive Lasso is used as the variable selection method.

Figure 4 shows that MCBs of Algorithm 2 have almost the same CR as that of Algorithm 1 even for different p when p is not large. However, MCBs of Algorithm 1 can be hardly constructed when p is large, while it is still feasible for Algorithm 2. Since we have defined the criterion of assessment for different methods, for the same width of MCBs, the best method has the largest CR. Therefore, the method with larger CR is better for a given width. Thus both of these two algorithms have excellent performance.

Since the two algorithms perform almost exactly alike, we could explore the computation time of these algorithms. The results of practical computation time is shown in Table 2. Note that computation time of Algorithm 1 increases dramatically as p increases, while Algorithm 2 remains. Especially when $p = 15$, Algorithm 1 takes nearly twenty hours, while Algorithm 2 needs no more than one second. It is not a big surprise since Algorithm 1 has to deal with a large number of iterations as we discussed in Section 2.

Although Algorithm 1 and Algorithm 2 have almost the same performance, Algorithm 2 takes much less time. Especially when $p > 15$, Algorithm 1 is even difficult to process, while Algorithm 2 computes still quickly even if when the number of predictors is large. Therefore we will use Algorithm 2 to do the remaining numerical studies.

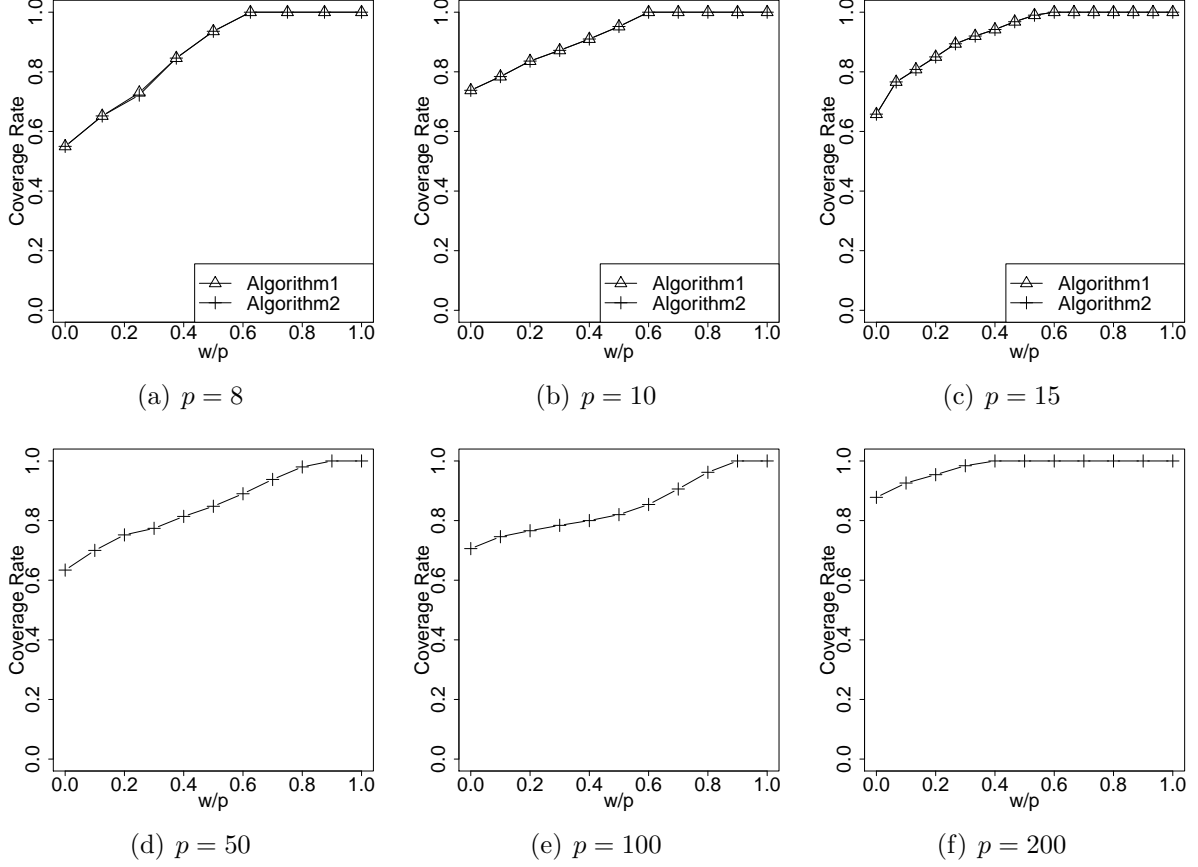


Figure 4: Performance of different algorithms for different p . The true model is $y = \sum_{j=1}^k x_j + \sum_{j=k+1}^p 0 \times x_j + \epsilon$, where $\epsilon \sim N(0, \sigma^2)$. In each figure, $n = 300$, $\rho = 0$, and $B = 1000$. The top row shows results of Algorithm 1 and Algorithm 2 for different scenarios: $p = 8, k = 3, \sigma = 1$ for (a), $p = 10, k = 4, \sigma = 1$ for (b), and $p = 15, k = 6, \sigma = 1$ for (c). The second row only shows results of Algorithm 2: $p = 50, k = 8, \sigma = 0.3$ for (d), $p = 100, k = 10, \sigma = 0.3$ for (e), and $p = 200, k = 12, \sigma = 0.3$ for (f), because the computation of Algorithm 2 is too high to perform as discussed in Section 3. Adaptive Lasso is used as the variable selection method in each bootstrap sample.

Table 2: Corresponding processing time (seconds) of procedures in Figure 4.

Method	$p = 8$	$p = 10$	$p = 15$	$p = 50$	$p = 100$	$p = 200$
Algorithm 1	22.37	207.64	67750.81	—*	—	—
Algorithm 2	0.12	0.16	0.41	5.01	22.70	201.43

* The value here cannot be acquired.

5.2 Performance of MCBs as $n \rightarrow \infty$

Consider the same model as (15), where $k = 6$, $p = 15$, $\sigma = 1$, $\rho = 0$ and $B = 1000$. We increase sample size gradually from 200 to 400 and explore the performance of MCBs in Figure 5. Note that MUC obviously arches towards the upper left corner as sample size increases. It makes sense because as sample size increases there will be less unique bootstrap models. Thus the variation of these models becomes small, meanwhile width of MCBs tends to get small. Therefore for the same width, the MCBs of larger sample size will tend to have larger CR and its MUC arches more towards the upper left. When the sample size is large enough ($n \rightarrow \infty$), the Adaptive Lasso will select only the true model according to the “oracle” property. The bootstrap models will be the same and MCBs will contain only the true model, thus $P(\hat{m}_L = m^* = \hat{m}_U) = 1$.

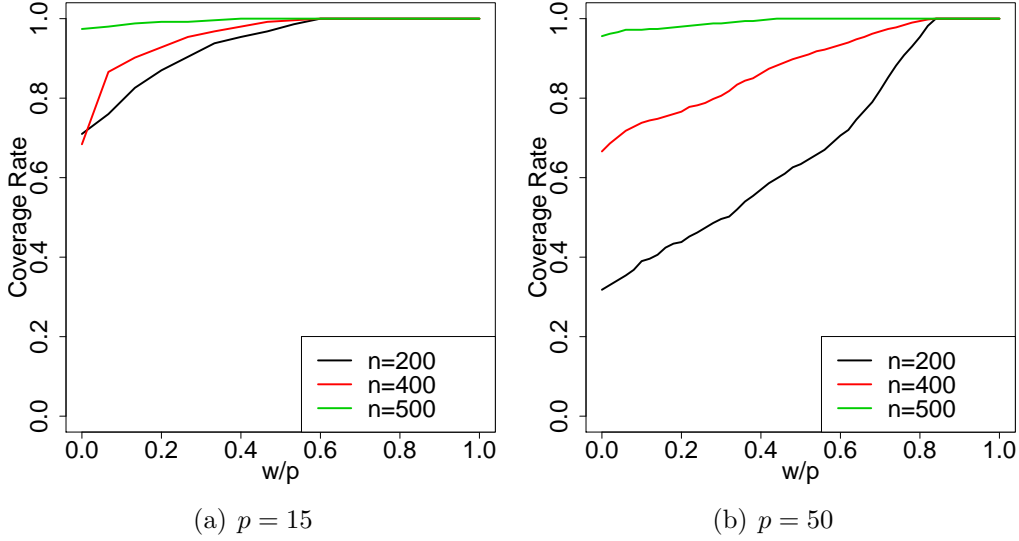


Figure 5: Performance of MCBs for different sample sizes. The true model is $y = \sum_{j=1}^k x_j + \sum_{j=k+1}^p 0 \times x_j + \epsilon$, where $\epsilon \sim N(0, 0.3^2)$. In the left panel $p = 15, k = 6$ while $p = 50, k = 8$ for the right panel. $B = 1000$ and $\rho = 0$ for both two settings. Adaptive Lasso is used as the variable selection method in each bootstrap sample.

5.3 Evaluate Different Variable Selection Methods based on MUC

One of the greatest advantages of MCBs is to provide a framework to assess uncertainty of different variable selection methods. We consider two settings. The first setting is linear regression: $y = \sum_{j=1}^5 x_j + \sum_{j=6}^{12} x_j * 0 + \epsilon$. In this setting we consider two scenarios: $\epsilon \sim \text{Normal}(0, 1)$ and $\epsilon \sim \text{Laplace}(0, \sqrt{1/2})$, so that they have the same mean and variance for ϵ . The other setting is logistic regression: $\log(\frac{p(Y=1)}{1-p(Y=1)}) = \sum_{j=1}^5 x_j + \sum_{j=6}^{12} x_j * 0$. $n = 300$, $\rho = 0$, and $B = 1000$ for both two settings. In the first setting, we compare different variable selection methods with different loss functions and penalties. Stepwise (BIC), Lasso, Adaptive Lasso, SCAD and MCP have the same loss function but with different penalties. While Lasso, LAD Lasso and SQRT Lasso have the same penalty but with different loss functions. According to the criterion of assessment, for a better method its MUC will arch more towards upper left. Figure 6 shows the performance of different variable selection

methods. For the normal distribution, Lasso, SCAD and MCP perform better than other methods whose loss functions are mean square, while SQRT Lasso and LAD Lasso perform worse than Lasso which have the same penalty. For the Laplace distribution, LAD Lasso and SQRT Lasso perform better than Lasso for the same penalty. It makes sense because LAD Lasso and SQRT Lasso are robust to heavy tail random noise. While Lasso, SCAD and MCP perform better than other methods for the same loss function.

Moreover, the proposed method can be easily extended to generalized linear model, such as logistic regression. In this setting we consider two scenarios: continuous and categorical predictors. We compare the performance of five different variable selection methods: Stepwise (BIC), Lasso, Adaptive Lasso, SCAD and MCP. Figure 7 shows the results for logistic variable selection models. Note that SCAD and MCP perform better in the continuous scenario while Lasso performs better in the categorical scenario. Therefore MCBs can be used as a tool to assess the performance of different variable selection methods for different data generating processes.

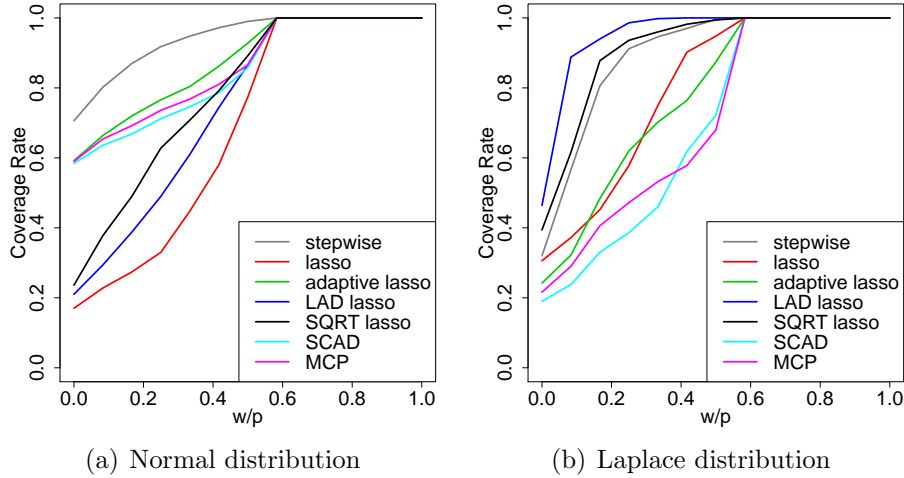


Figure 6: Performance of different variable selection methods based on MUC for different distributions. In this setting we consider the linear regression: $y = \sum_{j=1}^5 x_j + \sum_{j=6}^{12} x_j * 0 + \epsilon$, where $\epsilon \sim \text{Normal}(0, 1)$ for (a) while $\epsilon \sim \text{Laplace}(0, \sqrt{1/2})$ for (b). $n = 300$, $B = 1000$, and $\rho = 0$ for both two settings. Stepwise select variables based on BIC criterion. Tuning parameters of other variable selection methods are chosen based on 10-fold cross validation.

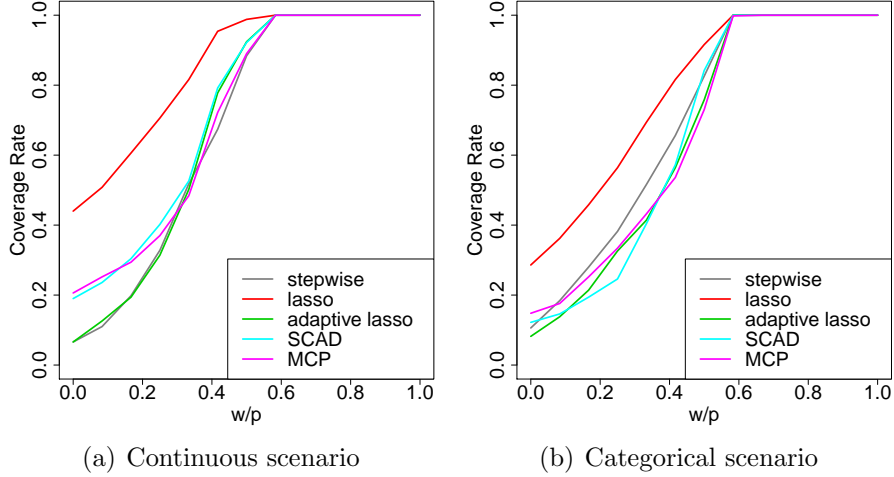


Figure 7: Performance of different variable selection methods based on MUC in logistic variable selection. We consider the model $\log(\frac{p(Y=1)}{1-p(Y=1)}) = \sum_{j=1}^5 x_j + \sum_{j=6}^{12} x_j * 0$, where all predictors are continuous for (a) and X_5, X_{11}, X_{12} are binary while other predictors are continuous for (b). $n = 300$, $B = 1000$, and $\rho = 0$ for both two settings. Stepwise selects variables based on BIC criterion. Tuning parameters of other variable selection methods are chosen based on 10-fold cross validation.

5.4 Comparison between BCR and TCR

So far, all the comparisons are based on estimated coverage rate. However, our true goal is to maximize the true model coverage rate (TCR: $\text{TCR} = \Pr(\hat{m}_L \subseteq m^* \subseteq \hat{m}_U)$), CR is an approximate for TCR. Thus we will explore the relationship between CR and TCR. Since the model uncertainty curve is complete based on CR, we can compare MUC with TMUC ($\mathcal{P}_{TMUC} = \{(w/p, \text{CR}(w)), 0 \leq w \leq p\}$) to assess the approximation of CR for TCR. We consider the same model as Equation (15) and three scenarios: $p = 8, p^* = 3, n = 1000, \sigma = 10$; $p = 20, p^* = 8, n = 1000, \sigma = 10$; and $p = 20, p^* = 8, n = 300, \sigma = 6$, while for all the scenarios $B = 1000, \rho = 0$ and in order to compute the the TCR, 500 replications are sampled and then generate an MCBs for each. Figure 8 shows the results of MUC and TMUC. Note that although the bootstrap coverage rate is a little different from the TCR at different widths, the MUC approaches the TMUC very closely in all the three scenarios.

Thus we can use the MUC to replace the TMUC.

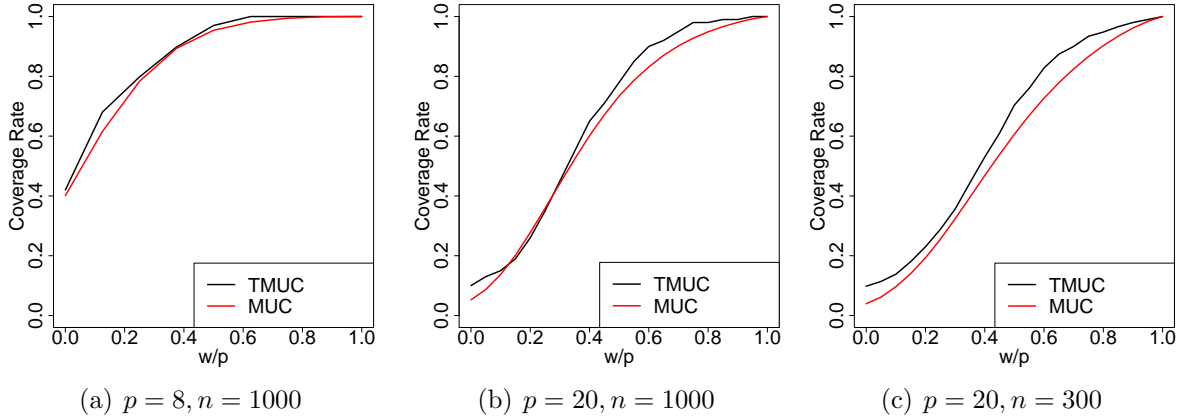


Figure 8: Comparison of MUC and TMUC. The true model is $y = \sum_{j=1}^k x_j + \sum_{j=k+1}^p 0 \times x_j + \epsilon$, where $\epsilon \sim N(0, \sigma^2)$. We consider three scenarios: $p = 8, n = 1000, \sigma = 10$ for (a), $p = 20, n = 1000, \sigma = 10$ for (b), and $p = 20, n = 300, \sigma = 6$ for (c). While in all three scenarios, $B = 1000$ and 500 replications are sampled to compute the TCR. Adaptive Lasso is used as the variable selection method.

6 Conclusions and Final Remarks

This paper introduces MCBs for variable selection of linear model and proposes a fast algorithm which can also easily extend for massive data sets. MCBs which consist of a LBM and a UBM contain the true model at a given confidence level, it extends the notion of confidence interval to the variable selection problem. MCBs can be used as a model selection diagnostic tool by comparing the model with the LBM and the UBM. One attractive advantage of MCBs is that it provides a platform to assess the uncertainty of different variable selection methods. Moreover, MCBs are constructed based on nonparametric bootstrap, it means that the method does not rely on the assumption of distribution.

MCBs provide more insights of existing variable selection methods and a deep understanding of original data sets. MCBs provide another framework for statistical inference in the contest of variable selection. Due to the duality of confidence interval and hypothesis

testing, MCBs can be used for hypothesis testing in model selection. Rather than blindly relying on a single model without knowing the credibility, MCBs yield two bounds for models that captures the uncertainty of these models.

Another advantage of MCBs is that it is not restrict to linear model, due to the properties of nonparametric bootstrap it can be extended to other popular models: such as time series, regression tree, etc. For example, it can provide an confidence interval for AR (average regression) models or MA (moving average) models. It can also provide an confidence interval for the rules in regression tree.

In this paper we assume that MCBs consist of only a LBM and a UBM. However, MCBs may contain multiple LBMs which are not statistically significantly different. For example, the MCBs with a LBM: $\{1, 2, 3\}$ and another MCBs with a LBM: $\{6, 7, 8\}$, both of these MCBs have the same UBM. Although the LBMs are different, these MCBs may have the same CRC. In this case these two MCBs are equivalent and thus we have to consider the situation of multiple LBMs in one MCBs. Similarly, MCBs may also have multiple UBMs. We leave these works for future research.

References

- C. De Mol, E. De Vito, and L. Rosasco. Elastic-net regularization in learning theory. *Journal of Complexity*, 25(2):201–230, 2009.
- B. Efron, T. Hastie, I. Johnstone, R. Tibshirani, et al. Least angle regression. *The Annals of statistics*, 32(2):407–499, 2004.
- J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96(456):1348–1360, 2001.
- D. Ferrari and Y. Yang. Confidence sets for model selection by f-testing. *Statistica Sinica*, 25:1637–1658, 2015.
- P. R. Hansen, A. Lunde, and J. M. Nason. Choosing the best volatility models: The model confidence set approach*. *Oxford Bulletin of Economics and Statistics*, 65(s1):839–861, 2003.
- P. R. Hansen, A. Lunde, and J. M. Nason. Model confidence sets for forecasting models. Technical report, Working Paper, Federal Reserve Bank of Atlanta, 2005.
- P. R. Hansen, A. Lunde, and J. M. Nason. The model confidence set. *Econometrica*, 79(2):453–497, 2011.
- J. Huang, S. Ma, and C.-H. Zhang. Adaptive lasso for sparse high-dimensional regression models. *Statistica Sinica*, pages 1603–1618, 2008.
- D. Jiang, J. Huang, and Y. Zhang. The cross-validated auc for mcp-logistic regression with high-dimensional data. *Statistical methods in medical research*, 22(5):505–518, 2013.
- C. Lindsey, S. Sheather, et al. Variable selection in linear regression. *Stata Journal*, 10(4):650, 2010.

- L. Meier, S. Van De Geer, and P. Bühlmann. The group lasso for logistic regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(1):53–71, 2008.
- J. D. Samuels and R. M. Sekkel. Forecasting with many models: Model confidence sets and forecast combination. Technical report, Bank of Canada Working Paper, 2013.
- H. Shimodaira. An application of multiple comparison techniques to model selection. *Annals of the Institute of Statistical Mathematics*, 50(1):1–13, 1998.
- N. Simon, J. Friedman, T. Hastie, and R. Tibshirani. A sparse-group lasso. *Journal of Computational & Graphical Statistics*, 22(2):231–245, 2013.
- R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.
- B. Yang, S.-G. Wang, and Y. Bao. Efficient local aadt estimation via scad variable selection based on regression models. In *Control and Decision Conference (CCDC), 2011 Chinese*, pages 1898–1902. IEEE, 2011.
- Y. Yang and H. Yang. Eigenvalue condition and model selection consistency of lasso. *arXiv preprint arXiv:1502.01798*, 2015.
- M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006.
- H. H. Zhang and W. Lu. Adaptive lasso for cox’s proportional hazards model. *Biometrika*, 94(3):691–703, 2007.
- H. Zou. The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429, 2006.

H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.